Discrete region competition for unknown numbers of connected regions.

Cardinale, J ; Paul, G ; Sbalzarini, I F

Abstract: We present a discrete, unsupervised multi-regioncompetition algorithm for image segmentation over different energy functionals. The number of regions present in an image does not need to be known a priori, nor their photometric properties. The algorithm jointly estimates the number of regions, their photometries, and their contours. The required regularization is provided by defining a region as a connected set of pixels. The evolving contours in the image are represented by computational particles that move as driven by an energy-minimization algorithm. We present an efficient discrete algorithm that allows minimizing a range of well-known energy functionals under the topological constraint of regions being connected components. The presented framework and algorithms are implemented in the open-source Insight Toolkit (ITK) image-processing library.

DOI: https://doi.org/10.1109/TIP.2012.2192129

Discrete region competition for unknown numbers of connected regions

Janick Cardinale, Grégory Paul, Ivo F. Sbalzarini, *Member, IEEE*

Abstract

We present a discrete, unsupervised multi-region-competition algorithm for image segmentation over different energy functionals. The number of regions present in an image does not need to be known *a priori*, nor their photometric properties. The algorithm jointly estimates the number of regions, their photometries, and their contours. The required regularization is provided by defining a region as a connected set of pixels. The evolving contours in the image are represented by computational particles that move as driven by an energy-minimization algorithm. We present an efficient discrete algorithm that allows minimizing a range of well-known energy functionals under the topological constraint of regions being connected components. The presented framework and algorithms are implemented in the open-source Insight Toolkit (ITK) image-processing library.

Index Terms

Region competition, multi-region segmentation, energy-based segmentation, discrete level set, deconvolution, digital topology, connected component, topological constraint

I. Introduction

Multi-region image segmentation partitions a digital image domain $\Omega \subset \mathbb{N}^d$ (here the dimension $d = 2$ or 3) into a background region X_0 and $(M-1) > 0$ disjoint foreground regions X_i, $i = 1, \ldots, M-1$, bounded by contours or surfaces[1] Γ_i, $i = 1, \ldots, M-1$. A large class of segmentation algorithms can

J. Cardinale, G. Paul, and I. F. Sbalzarini are with the MOSAIC Group at ETH Zurich and the Swiss Institute of Bioinformatics in Zurich, Switzerland.

[1]The term "contour" is used throughout this article to mean either "outline" (2D) or "surface" (3D). Similarly, we use "pixel" to mean either "pixel" (2D) or "voxel" (3D).

be interpreted as (local) minimizers of certain energy functionals. Energy-based segmentation allows accounting for prior knowledge about the imaged objects and about the image-formation process, including knowledge about the topology of objects [1] or their shapes.

A prominent and well-studied example of a segmentation energy is the functional proposed by Mumford and Shah [2] (MS). For binary images and images containing two regions (foreground and background) of constant intensities, this functional is notably known from the works of Chan and Vese [3] (CV). In the CV case, optimal solutions can be found using graph-cut algorithms [4], [5]. For images containing a priorly known number of constant-intensity regions, the MS energy can be convexified allowing globally optimal solutions to be computed efficiently [6], [7].

Regions in an image are usually defined through their intensities or other photometric or texture features. Multi-region segmentation then amounts to grouping pixels according to their features. Regions may hence comprise several disconnected sets of pixels, and the number of regions (i.e., the number of feature groups) frequently needs to be imposed, penalized, or learned *a priori*. Here we introduce the constraint that a foreground region has to consist of a connected set of pixels. This is motivated by the observation that frequently discrete physical objects are represented in an image. Moreover, topological constraints can be evaluated using local information only, whereas region-number priors require global information. The present definition of a region regularizes the problem of estimating the number of regions jointly with their photometric features and contours. We extend concepts from digital topology to enforce the topological region definition, and we present an efficient discrete energy minimization algorithm that can locally minimize a range of well-known energy functionals under this hard constraint.

We focus on images that contain unknown numbers of regions of not necessarily homogeneous intensities. The corresponding energy functionals are often non-convex, as for example the piecewise smooth MS energy [2], [8] or a deconvolving energy [9]. Segmentations are found as regularized local minimizers, formalized in the framework of *deformable models*. Deformable models entail an evolvable (deformable) continuous or discrete representation of the contours Γ_i. Local optimization is done by iterated perturbation of an initial contour Γ_0 such as to locally minimize the energy.

In unsupervised multi-region segmentation the number of regions, their photometric features, and their contours are to be jointly estimated from the image. This requires additional regularization on top of the usual smoothness priors. Most multi-region methods use region-number priors (e.g., [10], [11]). Alternatively, a length/area balancing term can be used [12]. Brox and Weickert proposed recursive splitting of regions into pairs of sub-regions such as to minimize an energy that includes a region-number penalty [10]. A separate level set is evolved for each region. In order to prevent regions from

overlapping, an additional penalization term is introduced into the energy functional. The number of level functions that need to be evolved is reduced to one in unsupervised region-competition methods [13]. A representation with only one level function, however, cannot capture multiple contours touching in one point. Other multi-region segmentation methods impose a fixed number of regions (or an upper bound on it) that is often learned prior to contour evolution using, e.g., pixel-feature clustering or model selection. This is for example the case in multi-phase level sets, which evolve $\log_2 M$ level functions in order to segment a fixed number of M regions [14]. Besides the increased computational cost of evolving multiple level functions, undefined statistics from empty regions may hamper the evolution [10]. Mansouri *et al.* [15] presented a multi-region-competition [16] implementation where the contours are implicitly represented by multiple level functions. Lie *et al.* represented multiple regions using a single level function that converges to a piecewise constant function indicating the different regions [17]. Homeomorphic level sets prevent topological changes during energy minimization [18].

Discrete implicit methods directly operate on the discrete constituents of a digital image, such as pixels or voxels. They switch the region labels of pixels in order to minimize an energy functional. Song and Chan introduced a fast discrete level-set method for the two-region piecewise constant CV model [19]. He and Osher generalized this method to an arbitrary, but priorly known number of piecewise constant regions [20] and related the approach to topological derivatives [21]. Yu *et al.* optimized a two-region piecewise smooth image energy using a discrete level function on a lattice [22]. Fast discrete level-set methods have been used for real-time tracking of a known, fixed number of regions [23] and for fast approximate surface evolution [24]. Graph min-cut algorithms [4] are efficient combinatorial optimizers for discrete problems with theoretical performance guarantees, both for priorly known numbers of regions [4] and for unknown numbers of regions using a region-number penalty [11].

Here we replace the prior or penalization on the region number (or its upper bound) by the topological constraint that foreground regions have to be connected components. Together with an efficient discrete contour evolution algorithm that accounts for this constraint, this constitutes the main contribution of the present work. We present an implementation of a versatile discrete-contour multi-region-competition algorithm in 2D and 3D, inspired by discrete level sets [23]. The algorithm is based on the idea of using computational particles to represent the evolving contour and is able to segment a priorly unknown and arbitrary number of connected regions. Regions are dynamically fused and split during energy minimization. This enables jointly estimating the number of connected regions in an image, their photometric features, and their contours. We use digital topology to provide optional control over region splits and merges during contour evolution. The topological constraint for foreground regions to

be connected components, however, is always present.

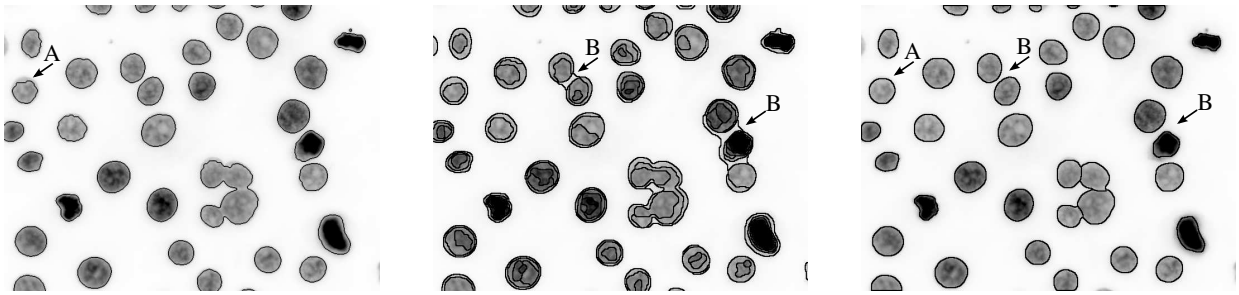
We demonstrate the applicability of the present method to three well-known segmentation energy functionals: The first energy describes images containing an unknown number of regions where each region has a different, but constant (homogeneous) intensity. The energy is regularized using a penalty on the approximated length of the overall contour. The second energy extends this model to account for regions containing piecewise smooth intensity distributions. The third energy extends explicit deconvolving active contours [9] to handle topological changes during energy minimization and to arbitrary dimensions. This renders the method less sensitive to the topology of the initial segmentation.

The computational cost of the present algorithm mainly depends on the energy functional to be minimized. For piecewise constant and piecewise smooth image models it scales linearly with the number of particles used to represent the contour and is independent of the size of the image. The present algorithm is implemented as an image filter in the Insight Toolkit (ITK) image-processing library [25] and is available as open source from the web page of the authors.

The remainder of this paper is organized as follows: In Sec. II we motivate the proposed definition of foreground regions and present an extension of digital topology to multiple regions. In Sec. III we present an efficient discrete algorithm for region-competition energy minimization under hard topological constraints. Section IV presents the applicability of the present framework to three well-known image models on both synthetic and real-world images in 2D and 3D, and compares its performance with that of a multi-label graph-cut minimizer [11]. Section V summarizes and discusses the results.

II. Digital geometry representation

In order to jointly estimate the number of regions, their photometric properties, and their contours, unsupervised multi-region segmentation energies typically include a region-number penalty [10], [11], [13] or a length/area regularizer [12]. Here we instead define a foreground (FG) region as a connected set of pixels in a certain digital geometry representation, amounting to a topological constraint. This definition is motivated threefold: (1) We wish that regions determined by a segmentation algorithm delineate different physical objects represented in an image (see Fig. 1). This frequently causes the problem of choosing an appropriate number of regions so as to avoid over- and under-segmentation (see arrows A and B in Fig. 1b). (2) It resolves the dependence between the number of regions and the regularization constant in the energy (see Fig. 2). (3) It can be evaluated using only local information, whereas region-number penalties require global information (see Sec. II-B).



(a) (b) (c)

Fig. 1. A motivation for defining FG regions as connected components. The image shows a collection of cell nuclei, which are distinct real-world objects (image: Dr. Prisca Liberali, University of Zurich). (a) Segmentation (black outlines) using graph cuts [4] to minimize the two-region CV energy. Due to their different intensities, not all nuclei are correctly delineated (see, e.g., arrow A). (b) Graph-cut segmentation minimizing a ten-region piecewise constant energy. It is not clear what number of regions to choose in order to avoid over-segmentation and fusion of objects (see arrows B). (c) Segmentation using the present algorithm constraining FG regions to be connected components. The algorithm finds 39 connected FG regions, corresponding to the 39 nuclei in the image.

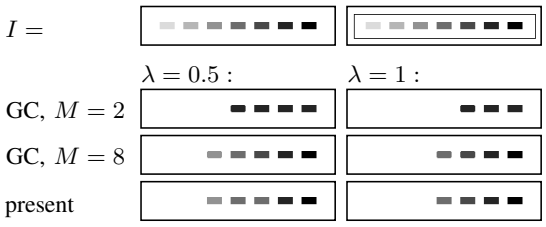


Fig. 2. Illustration of the dependence between the number of regions and the length-regularization coefficient λ in a piecewise constant image. 1st row: raw image I (left) and initialization for the present algorithm (right). 2nd and 3rd rows: resulting reconstructed images using graph cuts (GC) [4] with $M = 2$ and $M = 8$ regions, respectively. The lowest intensity that is detected depends on both M and λ. 4th row: present reconstruction when defining a FG region as a connected component. The result corresponds to the GC result with the ground-truth number of $M = 8$ regions. The lowest intensity detected only depends on λ.

We first present the digital geometry representation used here and then provide an extension of digital topology to multiple regions.

A. Digital geometry representation

1) Connectivity of regions: We constrain FG regions in the image to be represented by connected pixels. All void space between FG regions is represented by one and the same background (BG) region. Regions that can be captured by this representation must be larger than a single pixel. Consequently, regions cannot be connected via edges or corners of the pixel lattice. The FG regions are hence defined as face-connected neighborhoods, i.e., 4-connected in 2D and 6-connected in 3D. In the following we

refer to this type of connectivity as the *FG connectivity*. According to Jordan's theorem, the BG region then needs to be 8-connected in 2D and 18 or 26-connected in 3D [26]. Here we choose the (FG, BG)-connectivity pairs $(4, 8)$ and $(6, 26)$ for 2D and 3D, respectively.

2) Contour: The discrete contour Γ_i around FG region X_i, $i = 1, \ldots, M-1$, is defined by all pixels with at least one FG-connected neighbor belonging to a different region $X_j \neq X_i$, $j = 0, \ldots, M-1$. These contour points are part of the corresponding FG region, i.e., $\Gamma_i \subset X_i$, making all FG regions closed, connected subsets of Ω. The BG region is the open complement set $X_0 = \Omega \setminus \bigcup_{i=1}^{M-1} X_i$.

B. Multi-region digital topology

For images with one FG and one BG region, the concept of digital topology allows detecting if changing the region label of a point (pixel) changes the genus of either the FG or the BG region [1], [26]–[28].

We briefly introduce the notions of *connectivity*, *geodesic neighborhoods*, and *topological numbers*. For more details on these topics we refer to Refs. [1], [27], [28]. We then extend these concepts to multiple FG regions.

We adopt the notation and definitions from Refs. [26], [27], [29]. Digital topology is a binary concept that defines the FG X as a set of discrete points x and the BG as its complement $\bar{X}$, such that $X \cap \bar{X} = \emptyset$ and $X \cup \bar{X} = \Omega$. Both FG and BG have a certain connectivity. In 2D, two points are 4-connected if they share an edge and 8-connected if they share a corner. In 3D, two points are 6-connected if they share a face, 18-connected if they share an edge, and 26-connected if they share a corner. In order to avoid topological paradoxes, only the following combinations of FG and BG connectivities are admissible according to Jordan's theorem: $(n, \bar{n}) \in \{(4, 8), (8, 4), (6, 26), (26, 6), (6, 18), (18, 6)\}$.

The n-neighborhood $N_n(x)$ is the set of n-connected points adjacent to point x.

Definition 1. *Let $X \subset \Omega$. The geodesic neighborhood of order k of a point $x \in X$ is the set $N_n^k(x, X)$ defined recursively by:*

$$\begin{cases} N_n^1(x, X) = \{N_n(x) \backslash x\} \cap X \\ N_n^k(x, X) = N_m^1(x, X) \cap \bigcup \{N_n(y),\, y \in N_n^{k-1}(x, X)\} \end{cases}$$

with $m = 8$ in 2D and $m = 26$ in 3D.

Intuitively, the geodesic neighborhood $N_n^k(x, X)$ comprises all points $y \in N_m^1(x, X) \backslash x$ that are n-connected to x along a path that is not longer than k [27].

From this, a topological number can be defined as the number of n-connected components $\#C_n(\cdot)$ within a geodesic neighborhood:

Definition 2. *The topological numbers $T_n(x, X)$ relative to the point x and the set X are:*

$$\begin{aligned} T_4(x,X) &= \#C_4(N_4^2(x,X)) & T_8(x,X) &= \#C_8(N_8^1(x,X)) \\ T_6(x,X) &= \#C_6(N_6^2(x,X)) & T_{6^+}(x,X) &= \#C_6(N_6^3(x,X)) \\ T_{18}(x,X) &= \#C_{18}(N_{18}^2(x,X)) & T_{26}(x,X) &= \#C_{26}(N_{26}^1(x,X))\,. \end{aligned}$$

The notation $n = 6^+$ indicates that the dual connectivity $\bar{n}$ is 18, whereas the dual connectivity for $n = 6$ is 26.

Topological numbers are an efficient tool to characterize points in binary images. They can be computed from local information. For example, if $T_n(x, X) = T_{\bar{n}}(x, \bar{X}) = 1$, we know that changing the region label of point x does not change the genus of neither the FG, nor the BG. All points for which this is true are called *simple points*.

We use topological numbers to classify points also in the present multi-region framework by splitting the FG $X = \bigcup_{i=1}^{M-1} X_i$ into multiple disjoint sub-regions X_i. The BG region remains a single set $\bar{X} = X_0$.

Definition 3. *A point x is* foreground simple *(FG-simple) iff $T_n(x, X_i) = T_{\bar{n}}(x, \bar{X}_i) = 1$ for all $i > 0$.*

Intuitively, $T_n(x, X_i)$ is the topological number when considering all other regions $X_j, j \neq i$, to be part of the BG. Changing the region label of a FG-simple point does not change the genus of *any* FG region. For example, all contour points except (b,5), (c,5), and (d,4) in Fig. 3 are FG-simple.

This extended definition of FG simplicity allows distinguishing different topological events on the FG regions. In this framework, the topological constraint that FG regions have to be connected components can be interpreted as a hard penalty in the segmentation energy.

III. Energy minimization algorithm

We introduce a versatile region-competition mechanism inspired by discrete level set methods. In the present framework, minimization of an energy $\mathcal{E}$ uses a rank-based discrete optimizer that does not require information about the gradient of the energy functional. This is beneficial since the hard penalty introduced by the topological constraint on regions is not differentiable. We start by introducing the data structures and then describe the minimization algorithm used to perform topologically consistent contour evolution. The algorithm is designed with data locality and parallelism in mind.

A. Data structures

The present method relies on three main data structures: First, regions are identified using a *label function* (or *label image*) $L\colon \Omega \mapsto \mathbb{N}$ that maps a discrete space coordinate x to the region label currently assigned to that pixel. Contour pixels are assigned the negative label of the region they bound. This allows identifying contour points directly from the label image. The label of the BG region is fixed to 0.

Second, all points belonging to a contour are stored as computational particles. Each particle p is defined by its location x_p, i.e., the integer pixel coordinates of the corresponding contour point, and its properties. These properties are used to propagate the contour and are stored in a *particle data structure* containing:

- the currently assigned label $L(x_p)$ to avoid expensive lookups in the label image;
- the candidate label l' as the label that minimizes $\Delta\mathcal{E}_p$ among all other candidate labels;
- the change in energy $\Delta\mathcal{E}_p$ when changing the current label l to the candidate label l';
- lists with the particle indices of the parent and child points of p. Parents are all FG-connected points that belong to a different FG region. They are responsible for expanding the FG region they belong to. Children are all FG-connected points that belong to a different region, including the BG.
- the count r of parents with label l'.

Third, we use a hash map $\Omega \mapsto \mathcal{C}$ as an efficient data structure to iterate over the particles and to map space coordinates x to particle indices p. The hash map allows index lookups in $O(1)$.

B. Algorithm

We describe an algorithm that iteratively propagates the contour points (viz., the particles) of multiple regions over the image such as to locally minimize an energy functional under topological constraints on the FG regions. After initialization, the algorithm proceeds in iterations (see Algorithm 1), each of which comprising three steps: optimization, contour propagation, and topology processing.

Application-specific segmentation methods can be derived from the present algorithm by specifying a particular energy functional and a set of topological constraints. The former allows including prior knowledge about the image-formation process (e.g., the point-spread function of a microscope in deconvolving active contours [9]) and the morphology of the imaged objects. The latter allows including prior knowledge about whether FG regions are allowed to fuse or split (or both or none) during the energy minimization process [1], [27], [28]. Regardless of additional topological constraints on contour evolution, however, a FG region is always defined as a FG-connected component.

Algorithm 1 Discrete region competition

1: Initialization: Set up L and $\mathcal{C}$.
2: **repeat**
3: $\mathcal{M} = \mathcal{C}$
4: Optimization: see Algorithm 2
5: Contour propagation: see Algorithm 3
6: Topology processing: see Algorithm 4
7: **until** convergence

The input arguments to the algorithm are an energy functional $\mathcal{E}$, the image data I, and, since it is an iterative process, an initial segmentation L_0. Pixels in L_0 that have a special label f can be used to indicate forbidden regions. These regions are treated as a boundaries that are never penetrated by any contour, nor do they have an active contour themselves. In order to avoid boundary checking at the border of the image domain Ω, we initially pad the entire image by a layer of pixels with label f.

1) Initialization: (line 1 in Algorithm 1) All FG pixels with a neighbor of a different label are marked as contour points. For each contour point, a particle is generated and added to the hash map $\mathcal{C}$, where the corresponding space coordinate is the key of the map and the particle index its value.

2) Optimization: In the main loop (line 2 in Algorithm 1), we first copy the current set of particles $\mathcal{C}$ to $\mathcal{M}$. $\mathcal{M}$ is the candidate list containing all particles we consider moving to another region. We first attempt moving them to the BG by setting all candidate labels l' in $\mathcal{M}$ to 0 (line 2 in Algorithm 2). We then calculate for each particle p the energy difference $\Delta\mathcal{E}_p = \Delta\mathcal{E}(x_p, l \to 0) = \mathcal{E}(x_p, 0) - \mathcal{E}(x_p, l)$.

In the next step, we attempt growing the FG regions. To do so, all particles $p \in \mathcal{M}$ perform the following steps: All neighboring points that belong to a different region (including the BG) register p as a parent (line 4). Particles for contour points that do not yet exist (since their current label is 0) are created and added to $\mathcal{M}$ (line 6). All particles now know the set of pixels they could potentially move to, and the set of pixels they are attacked from.

The candidate label l'_q of q is set to the label of p if this is favorable in energy (lines 8–11). This means that if the candidate label of q is different from the label of p (else we increase the parent count r since this candidate label is supported by two or more parents, line 9), we set l'_q to the label l_p of the parent if $\Delta\mathcal{E}(x_q, l_q \to l_p) < \Delta\mathcal{E}(x_q, l_q \to l'_q)$. In addition, we remove particles with $\Delta\mathcal{E} \geq 0$ from the candidate list (line 13).

Algorithm 2 Optimization

for all $p \in \mathcal{M}$ **do**
  set parent flag; $l'_p = 0; \Delta\mathcal{E}_p = \Delta\mathcal{E}(x_p, l \to 0)$.
  for all $\{q \mid x_q \in \{N_n^1(x_p, L \neq l_p)\}, l_q \neq f\}$ **do**
    set child flag of q; register p in q's parent list; register q in p's daugther list
    if $q \notin \mathcal{M}$ **then**
      add q to $\mathcal{M}$; Set $l_q = 0$; $r_q = 1$; $l'_q = l_p$; $\Delta\mathcal{E}_q = \Delta\mathcal{E}(x_q, 0 \to l'_q)$
    else
      if $l_p = l'_q$ **then**
        $r_q = r_q + 1$
      else if $\Delta\mathcal{E}(x_q, l_q \to l'_q) > \Delta\mathcal{E}(x_q, l_q \to l_p)$ **then**
        $l'_q = l_p$
construct $\mathcal{G}$ from $\mathcal{M}$
$\mathcal{M} = \mathcal{M} \backslash \{p : \Delta\mathcal{E}_p \geq 0\}$

While each individual move in $\mathcal{M}$ is guaranteed to decrease the overall energy, this may not be true for several moves performed simultaneously. This property is inherent to discrete contour-propagation methods and can cause contour and energy oscillations. We therefore monitor the history of the contours and halve the percentage of accepted moves whenever the contours do not propagate anymore. This amounts to reducing the step size in a rank-based optimization scheme. Unless the algorithm has already converged, the step size eventually reduces to 1, i.e., only a single move from $\mathcal{M}$ is executed in each iteration. This guarantees that the energy can only decrease from then onward, and the algorithm hence converges to a local minimum of $\mathcal{E}$.

3) Contour propagation: The set of moves that will be executed simultaneously needs to be selected according to the topological and causal constraints. Simply executing all minimum-energy moves determined in the optimization step could lead to violations of the topological constraints. Only contour points that are not FG-simple are allowed to cause a topological change in any FG region.

Topological violations can arise from the fact that moves at iteration t may depend on moves at iteration $t+1$. This is illustrated in Fig. 3 for the points $(\mathrm{d}, 2)$ and $(\mathrm{c}, 3)$. Whether region A is allowed to propagate to pixel $(\mathrm{d}, 2)$ without disconnecting depends on the label of pixel $(\mathrm{c}, 2)$ at iteration $t + 1$. The move at iteration t is only valid if pixel $(\mathrm{c}, 2)$ will still belong to region A at iteration $t + 1$. But $(\mathrm{c}, 2)$ has

Algorithm 3 Contour propagation

1: find maximal-connected subgraphs $\mathcal{G}_k$ of $\mathcal{G}$.
2: **for all** $\mathcal{G}_k = \{V_k, E_k\}$ **do**
3: sort V_k according to $\Delta\mathcal{E}$
4: **for all** $p \in V_k$ with $\mathcal{E}_p < 0$ **do**
5: **if** conditions C1, C2, and C3 are true **then**
6: $\forall$ children q with $l'_q = l_p$: $r_q = r_q - 1$.
7: **else**
8: $\mathcal{M} = \mathcal{M} \backslash p$

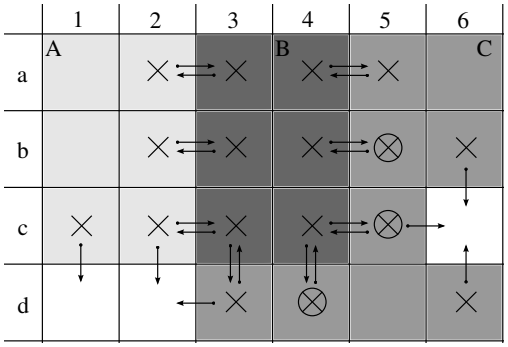


Fig. 3. Illustration of 3 adjacent FG regions A (light gray), B (dark gray), and C (gray) in 2D. Points in the background region are white. Particles are shown as crosses. Points without a particle are interior points; they are not FG-connected to any other region. The arrows point from parents to the corresponding children. The circles indicate non-foreground-simple points; interior points are not considered. See main text for details about the algorithm.

a parent at $(\mathrm{c}, 3)$, proposing it to join region B. This point at $(\mathrm{c}, 3)$ in turn is a candidate for label C through the parent at $(\mathrm{d}, 3)$. Situations like this can induce topological dependence chains of arbitrary length. We identify the set of moves that are topologically dependent by constructing an undirected graph $\mathcal{G} = \{V, E\}$ (line 12 in Algorithm 2). The vertices V correspond to particles and the undirected edges E to parent–child relationships. Topologically dependent sets are then given by the maximal-connected subgraphs $\mathcal{G}_k$ of $\mathcal{G}$. The maximal-connected subgraph in the example of Fig. 3 contains the vertices $\{(\mathrm{c}, 2), (\mathrm{c}, 3), (\mathrm{d}, 2), (\mathrm{d}, 3)\}$.

The contour is then propagated by selecting all compatible moves in $\mathcal{G}_k$ such as to minimize the sum of their energy differences. This is done independently for each subgraph $\mathcal{G}_k$. In order to avoid enumerating all compatible moves, we use a sub-optimal heuristic (Algorithm 3). This starts by sorting the vertices V_k of each subgraph by ascending $\Delta\mathcal{E}$ (line 3 in Algorithm 3) and purging all invalid moves from $\mathcal{M}$ in this order. Moving particle p is valid if it fulfills all of the following conditions (line 5):

C1: if p is a child, its parent count is $r_p \geq 1$;

C2: if p is a parent, all of its children q that have already been accepted as a move have $r_q > 1$;

Algorithm 4 Topology processing

1: change = true
2: **while** change **do**
3: change = false
4: **for all** $p \in \mathcal{M}$ **do**
5: **if** x_p is FG-simple **then**
6: Update data structures: Algorithm 5(x_p)
7: change = true
8: **for all** $p \in \mathcal{M}$ **do**
9: **if** holes are disallowed AND $\left(T_n(x_p, L = l'_p) \geq 2 \text{ OR } T_{\bar{n}}(x_p, L \neq l_q) \geq 2\right)$ **then**
10: **next** p
11: **if** $T_n(x_p, L = l_p) \geq 2$ **then**
12: **next** p **if** splits are disallowed
13: store the seed set $\mathcal{S} = \{N_n^1(x_p, L = l)\}$.
14: Update data structures: Algorithm 5(x_p)
15: **for all** $X_i \sim X_j$, $i, j = 1, \ldots, M-1$, **do**
16: **if** fusions are allowed AND region merging criterion is true **then**
17: merge regions X_i and X_j and add seed to $\mathcal{S}$
18: Recompute L using flood fill from seeds $\mathcal{S}$

C3: if p is a parent, at least one of its children is not yet accepted or has a candidate label $l'_q \neq l_p$.

C1 ascertains that the particle is connected to the propagating region. C2 ensures that no child of this particle would lose connection to the propagating region if this parent changed its label. C3 prohibits moves to interior points. Valid moves for a parent p reduce the parent counts of all its children q with $l'_q = l_p$ (line 6).

4) Topology processing: We detect and account for topological changes in the FG regions using concepts from digital topology [1], [23], [26]–[28]. The BG region is allowed to change its topology arbitrarily. A genus change in a FG region can be a *split* of the region into several regions, a *fusion* of two or more regions into one, or the introduction of a *hole* into a region.

Splits and the introduction of holes are detected using the FG topological number. If $T_n(x_p, \{y : L(y) = l'_p\}) \geq 2$ or $T_{\bar{n}}(x_p, \{y : L(y) \neq l_q\}) \geq 2$, changing the label of particle p to l'_p introduces a hole

Algorithm 5 Data-structure update

1: $L(x_p) = l'_p$.
2: **if** $l_p \neq 0$ **then**
3: Add $x' \in N_n^1(x_p, l_p)$ to $\mathcal{C}$; $L(\vec{x'}) = -l_{p'}$
4: **if** $l_{p'} \neq 0$ **then**
5: Remove all interior points in $x' \in N_{\bar{n}}^1(x_p, l'_p) \cup x_p$ from $\mathcal{C}$ and set $L(\vec{x'}) = |l'_p|$.
6: **else**
7: $\mathcal{C} = \mathcal{C} \backslash x_p$

in the children of region l_q (line 9 in Algorithm 4). Similarly, if the FG topological number for the label l_p is larger than 1, the corresponding region splits, unless splits are disallowed by the user (lines 11–12).

If region fusions are allowed, all competing pairs of regions (indicated by $\sim$ in line 15) undergo a region-merge check (line 16). In principle, this check depends on the energy functional $\mathcal{E}$. Different energy-independent merging criteria, however, have been introduced based on region statistics [16], [30], [31]. Here we use the symmetric Kullback-Leibler merging criterion [30] based on measuring the similarity between the empirical intensity distributions P_{X_i} and P_{X_j} of the two regions X_i and X_j, $i, j > 0$. The regions fuse if

$$D_{\mathrm{KL}}(P_{X_i}||P_{X_i \cup X_j}) + D_{\mathrm{KL}}(P_{X_j}||P_{X_i \cup X_j}) < \theta\,, \tag{1}$$

where $D_{\mathrm{KL}}(\cdot||\cdot)$ is the Kullback-Leibler divergence between the two distributions in the argument. The merging threshold θ is a free parameter of the method. For $\theta = 0$, regions are prevented from fusing.

Whenever region labels change due to splits or fusions, a seeded flood fill in L is performed to identify the new connected components. For fusions, the seed point is one of the pixels where the regions touch. For splits, all FG points neighboring points where the regions were last in contact are seeds. The points of last contact are easily found as those that are not FG-simple (line 13). If a seed point moves to a different region, another point in its geodesic neighborhood of order 1 becomes the new seed. The flood fill (line 18) then reconstructs the label image L.

5) Data-structure update: During topology processing, moves that do not induce topological violations are executed and the data structures updated (Algorithm 5 called from lines 6 and 14 of Algorithm 4). The labels of the corresponding points are changed to the respective candidate labels, and the label image is updated accordingly (line 1 in Algorithm 5). These changes may causes the creation of new contour points, the particles of which are added to the hash map $\mathcal{C}$ (line 3). Similarly, the particles from pixels

that newly became interior points are removed from $\mathcal{C}$ (lines 5 and 7).

IV. Benchmarks and Applications

We demonstrate the capabilities and limitations of the proposed topological region prior and minimizer by applying them to synthetic benchmark images with three different energy models. In each case, we also illustrate the practical applicability of the method to real-world images and provide computational timings. All times reported have been measured on a single 2.67 GHz Intel i7 core with 4 GB RAM using the Intel C++ compiler (v. 12.0.2). All test cases and results are summarized in Table I. As a benchmark, we compare with iterated extended α-expansions with label costs as a region-number penalty [11]. We use an 8-neighborhood with edge weights following the Cauchy-Crofton formula [5]. The α-expansions are iterated in a PEARL-like manner in order to solve the joint estimation problem of region numbers, intensities, and contours [32]. We choose this graph-cut-based benchmark algorithm, referred to as GC below, since it is also discrete and provides good theoretical performance guarantees. The corresponding source code was obtained from `http://vision.csd.uwo.ca/code/`.

The first two benchmarks consider discrete versions of the MS energy [2]

$$\mathcal{E}(u, I;\, \eta, \lambda) = \sum_{i=0}^{M-1} \left(\sum_{x \in X_i} (u(x) - I(x))^2 + \eta \sum_{x \in X_i} |\nabla u|^2 + \lambda |\Gamma_i| \right) ,$$

where η and λ are regularization parameters, ∇ a discrete Nabla operator, and $\Gamma_0 = \emptyset$. Minimizing this functional amounts to finding a regularized piecewise smooth approximation u to the original image I, such that the total edge set $|\Gamma| = \sum_i |\Gamma_i|$ is minimized.

The terms in $\mathcal{E}$ generally fall into two categories: external and internal energy terms. External energy terms are responsible for data fidelity. They measure how likely it is that the current segmentation has produced the given image I. In the MS energy, the first sum over x represents the external energy. Internal energy terms are independent of the image I. They provide regularization and are often used to model prior knowledge about the noise process and the properties of the imaged objects, such as their shape, size, intensity, or texture.

In the following benchmarks we use the energy

$$\mathcal{E} = \mathcal{E}_{\text{data}} + \lambda \mathcal{E}_{\text{length}} + \alpha \mathcal{E}_{\text{merge}} \,. \tag{2}$$

The internal energies $\mathcal{E}_{\text{length}}$ and $\mathcal{E}_{\text{merge}}$ are described in the next sub-section. They include a regularization for the discrete contour length and a prior for region merging. All benchmarks use these same internal energies, but different external energies. For the external energy $\mathcal{E}_{\text{data}}$, we first consider the MS energy

with $\eta \to \infty$, resulting in a piecewise constant approximation of I. While we consider arbitrary numbers of FG regions, the binary case with two regions would correspond to the CV model [3]. The second benchmark considers a piecewise smooth approximation with an arbitrary number of regions. In the third benchmark we extend deconvolving active contours [9] to not priorly known numbers of regions. This is done by augmenting the piecewise constant multi-region energy by a convolution operation in the image-formation model.

A. Internal energy

1) Contour length regularization: The contour length energy is given by $\mathcal{E}_{\text{length}} = |\Gamma|$. In continuous active contour representations, such as level-set methods, the contour length can easily be computed. In discrete methods, however, it needs to be approximated from the discrete contour pixels using concepts from digital geometry.

Zhu and Yuille argued [16] that blurring an image with a Gaussian filter has similar effects as including a length-regularization term in the energy functional. One problem with this approach, however, is that edges get smoothed. Also, spurious intensity fluxes across close regions can be a problem since they change the mean intensities of these regions.

Another approximation used in Refs. [19], [22], and in techniques based on the Ising model, counts the number of region changes on the pixel grid. While this approach is computationally very efficient, it causes the regions to tend to polygonal shapes instead of developing smooth contours [33] [22]. Also, the contour generally does not evolve smoothly, due to the discrete objective function. Shi and Karl [23] hence smoothed the contour of a discretized level function using a Gaussian kernel, followed by a re-discretization step. A drawback of this approach is that the smoothing is not represented in the energy functional. The resulting tradeoff between regularity and data fidelity is hence difficult to assess [33].

This has been addressed by Kybic and Kratky [33], who proposed a regularizing flow for discrete level-set methods that approximates the local curvature κ as

$$\kappa(x) = C \left(\frac{|S_x^{R_\kappa} \cap X_{|L(x)|}|}{|S^{R_\kappa}|} - \frac{1}{2} \right) ,$$

with $S_x^{R_\kappa}$ a hypersphere of radius R_κ centered at x and $|S^{R_\kappa}|$ its volume. C is a constant that depends on the dimension d and on R_κ. Here we adopt this approach, exploiting the fact that curvature regularization is equivalent to contour-length regularization.[2] Unless otherwise stated we use $R_\kappa = 4$, which is found

[2]This is seen by applying variational calculus to $\sum_{i>0} \lambda |X_i| = \lambda \sum_{i>0} \int_{\Gamma_i} \mathrm{d}s$.

to provide a good tradeoff between regularity and resolution. We directly add the curvature-regularizing flow to the $\Delta\mathcal{E}$ of the particles. The direction of the flow is given by the outward normal on the contour. We adapt the sign of κ to account for the direction of the flow: for expanding regions, κ is subtracted from the energy difference, for shrinking regions it is added to it.

2) Region-merging prior: Since we define regions as connected components, they can naturally split during the energy-minimization process, provided these topological changes are permitted by the user. The criterion for regions to merge as introduced in Eq. 1 can be formulated as a hard region-merging penalty in the energy functional:

$$\mathcal{E}_{\text{merge}} = \sum_{(i,j)>0:X_i\sim X_j} H[D_{\text{KL}}(P_{X_i}||P_{X_i\cup X_j}) + D_{\text{KL}}(P_{X_j}||P_{X_i\cup X_j}) - \theta]\,. \tag{3}$$

$H(\cdot)$ is the Heaviside distribution and $X_i \sim X_j$ indicates that X_i and X_j are FG-connected competing regions. Two regions merge if this is favorable for the overall energy. In order to reflect the discrete-event character of topological changes, the weight α of this contribution to the total energy in Eq. 2 is set to ∞.

B. Multi-region piecewise constant image model

We first consider images comprising an unknown and arbitrary number of connected FG regions with each region having a potentially different, but constant mean intensity.

1) External energy: The external energy in this case is given by the "cartoon limit" of the MS energy for $\eta \to \infty$. This results in a piecewise constant approximation of I. The resulting external energy term is:

$$\mathcal{E}_{\text{data}}^{\text{PC}} = \sum_{i=0}^{M-1} \left(\sum_{x\in X_i} (c_i - I(x))^2 \right). \tag{4}$$

M dynamically counts the number of regions during energy minimization. The scalar c_i is the estimated mean intensity in region i.

2) Implementation: The intensity estimates c_i are taken to be the mean intensities μ_i of the corresponding region [2]. They are updated on the fly whenever a pixel enters or leaves a region. This allows evaluating $\Delta\mathcal{E}_{\text{data}}$ from the data structures presented in Sec. III-A without ever computing the absolute energy. The overall algorithmic complexity thus is in $O(|\Gamma|R_\kappa^d)$, where $|\Gamma|$ is the number of particles and $O(R_\kappa^d)$ results from the memory lookups in the label image that are needed to evaluate $\Delta\mathcal{E}_{\text{length}}$.

TABLE I

TEST CASES BENCHMARKING THE PRESENT OPTIMIZER (PRESENT) AGAINST MULTI-LABEL GRAPH CUTS (GC) FOR DIFFERENT ENERGY FUNCTIONALS (PC: PIECEWISE CONSTANT; PS: PIECEWISE SMOOTH; DEC: DECONVOLVING) AND IMAGES. $\mathcal{E}_{\text{GT}}$ IS THE ENERGY OF THE GROUND-TRUTH SEGMENTATION FOR THE SYNTHETIC TEST CASES.

Optimizer	Initialization	Optimizer parameters	$\mathcal{E}_{\text{data}}$	Energy parameters	Final $\mathcal{E}$	$(\mathcal{E}-\mathcal{E}_{\text{GT}})/\mathcal{E}_{\text{GT}}$	Iterations	CPU time
Icecream PC 2D, 130×130, Fig. 4								
present	6×6 bubbles	$\theta = 0.2$, $R_\kappa = 4$	PC	$\lambda = 0.04$	71.42	4e-5	64	0.39s
GC	$M = 12$	labelcost = 5	PC	$\lambda = 0.04$	71.28	**-1e-3**	3	0.28s
GC	$M = 6$	labelcost = 0	PC	$\lambda = 0.04$	75.85	0.06	3	**0.09s**
Icecream PC 2D, 410×410								
present	8×8 bubbles	$\theta = 0.2$, $R_\kappa = 8$	PC	$\lambda = 0.04$	467.2	5.2e-3	110	7.34s
GC	$M = 12$	labelcost = 5	PC	$\lambda = 0.04$	464.3	**-1.0e-3**	5	8.18s
GC	$M = 6$	labelcost = 0	PC	$\lambda = 0.04$	760.8	0.63	3	**1.3s**
Icecream PC 3D, $100 \times 100 \times 100$								
present	$5 \times 5 \times 5$ bubbles	$\theta = 0.2$, $R_\kappa = 4$	PC	$\lambda = 0.04$	1863	5.5e-3	62	57s
GC	$M = 12$	labelcost = 5	PC	$\lambda = 0.04$	1844	**-4.4e-3**	5	76.9s
GC	$M = 6$	labelcost = 0	PC	$\lambda = 0.04$	1880	0.014	5	**38.5s**
Zebrafish embryo nuclei 3D, $512 \times 512 \times 39$, Fig. 6								
present	local maxima	$\theta = 0$, $R_\kappa = 2$	PC	$\lambda = 0.04$	*	-	44	7.3m
Bird, 481×32, Fig. 5a/b								
present	18×12 bubbles	$\theta = 0.5$, $R_\kappa = 8$	PC	$\lambda = 0.2$	*	-	83	**4.06s**
GC	$M = 5$	labelcost = 50	PC	$\lambda = 0.2$	*	-	9	8.81s
Icecream PS 2D, 130×130, Fig. 7								
present	5×5 bubbles	$\theta = 0.2$, $R_\kappa = 4$	PS	$\lambda = 0.04$, $\beta = 0.05$, $R = 8$	87.94	8.5e-4	71	**0.49s**
GC	5×5 bubbles	labelcost = 20.5	PS	$\lambda = 0.04$, $\beta = 0.05$, $R = 8$	87.87	**-5.3e-5**	9	10.2s
GC	3×3 bubbles	labelcost = 40	PS	$\lambda = 0.04$, $\beta = 0.05$, $R = 8$	87.87	**-5.3e-5**	8	3.47s
Icecream PS 3D, $100 \times 100 \times 100$								
present	$3 \times 3 \times 3$ bubbles	$\theta = 0.3$, $R_\kappa = 4$	PS	$\lambda = 0.04$, $\beta = 0.05$, $R = 8$	4618	6.7e-4	77	**4m**
GC	$M = 3$	labelcost = 20.5	PS	$\lambda = 0.04$, $\beta = 0.05$, $R = 8$	4615	**-2.7e-6**	4	12m
Zebrafish embryo germ cells 3D, $188 \times 165 \times 30$, voxel size = $(506 \times 506 \times 1500\,\text{nm})$, Fig. 8								
present	bounding box	$R_\kappa = 0.04$	PS	$\lambda = 0.08$, $\beta = 0.005$, $R = 4.5\mu\text{m}$	*	-	207	5.3m
Cloud 2D, 481×32, Fig. 9a/b								
present	18×12 bubbles	$\theta = 0.2$, $R_\kappa = 8$	PS	$\lambda = 0.2$, $\beta = 0.1$, $R = 30$	*	-	157	**57.77s**
GC	3×5 bubbles	labelcost = 175	PS	$\lambda = 0.2$, $\beta = 0.1$, $R = 30$	*	-	16	12.3m
Elephants 2D, 130×130, Figs. 5c/d and 9c/d								
present	18×12 bubbles	$\theta = 0.5$, $R_\kappa = 8$	PC	$\lambda = 0.2$	*	-	163	**11.26s**
GC	$M = 5$	labelcost = 50	PC	$\lambda = 0.2$	*	-	13	42.57s
present	18×12 bubbles	$\theta = 0.2$, $R_\kappa = 8$	PS	$\lambda = 0.2$, $\beta = 0.05$, $R = 30$	*	-	385	**25.57s**
GC	3×5 bubbles	labelcost = 175	PS	$\lambda = 0.2$, $\beta = 0.05$, $R = 30$	*	-	17	13.2m
Convolved artificial image 2D, 49×72, Fig. 10								
present	bounding box	$\theta = 0.2$, $R_\kappa = 4$	DEC	$\lambda = 0.04$	27.13	-5e-2	53	2.3s
Endosomes 2D, 512×386, Fig. 11								
present	local maxima	$\theta = 0.1$, $R_\kappa = 2$	DEC	$\lambda = 0.04$	*	-	41	32s

* Final energy not comparable due to different definitions of a region.

3) Benchmarks on synthetic data: Figure 4 illustrates the behavior of the present algorithm and of GC using the above energy functional on a synthetic image. The image contains 6 regions, each of which having a different, but constant mean intensity. The present algorithm is started with an initial segmentation far from the correct result and with a wrong number of initial regions (Fig. 4e). This demonstrates the capability of the algorithm to merge regions and to correctly delineate boundaries between touching regions. The total computational time used for this example is 0.39 s, despite the unfavorable choice of initial contours.

The evolution of the total, external, and internal energies for this case is shown in Fig. 4i. The present algorithm converges after 64 iterations. The circle symbols mark the time points at which fusions between two or more regions occurred. GC rapidly finds a solution with a slightly lower energy than the ground truth. This can be explained by the noise introducing spurious minima in the energy landscape. The CPU times until convergence are comparable for the two algorithms. In order to test how the results scale with image size, we also consider the same problem with the image zoomed (not padded) to 410×410 pixels, and with a 3D version of the image (see Table I). In all cases GC is sensitive to the initial number of regions (Figs. 4c and 4d) when using uniformly distributed initial region intensity estimates. With an initial number of $M = 12$ regions GC solves the problem with a CPU time comparable to the present algorithm; for $M = 6$ GC fails to find the correct segmentation. The GC implementation requires ≈ 1.76 GB of main memory for this 3D case; the present code uses ≈ 125 MB.

4) Application to real data: We assess the real-world applicability of the present algorithm by applying it to 2D natural-scene images from the Berkeley database [34] and to a 3D confocal fluorescence microscopy image of stained nuclei in a zebrafish embryo. The results are shown in Figs. 5 and 6. In Fig. 5 we visually compare with GC results; the energies, however, cannot be compared due to the different definitions of what constitutes a region. The nuclei in Fig. 6 are small enough to justify the model of constant intensity within each nucleus. Different nuclei, however, have different intensities, e.g., arrows A and B in Fig. 6a, benefitting from a multi-region segmentation approach. The final label image after 44 iterations is shown in Fig. 6b. For better visualization, the gray-scales are the region labels rather than the estimated intensities. An overlay of the original image and the final contours is shown in Fig. 6c for the region highlighted by the yellow area in Fig. 6b. Figure 6d shows the result when allowing region fusions, illustrating the effect of topological control during contour evolution.

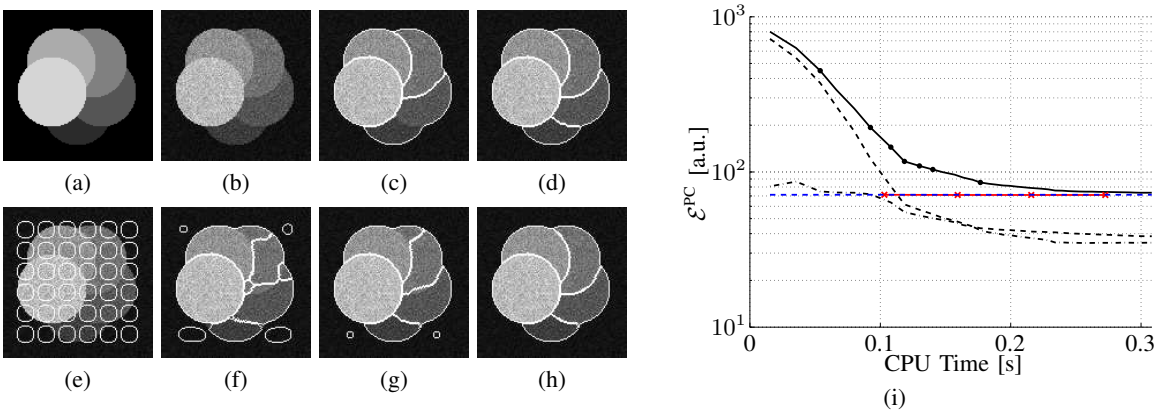
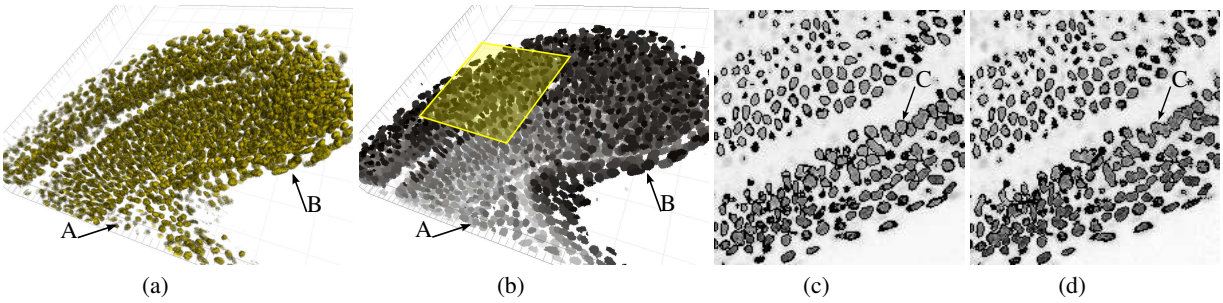


Fig. 4. Synthetic example using the energy $\mathcal{E}^{\mathrm{PC}}$. (a) Piecewise constant ground-truth image. (b) Ground-truth image corrupted with Poisson noise. The 5 FG regions correspond to peak signal-to-noise ratios (SNR) of 4, 5.25, 6.5, 7.75, and 9, respectively. (c) Final result from GC when initialized with the ground-truth number of $M = 6$ regions. The GC algorithm fails due to inaccurate estimates of the region intensities. (d) Correct GC result with 6 final regions when initializing with $M = 12$ regions. (e–h) Contour evolution at iterations 0, 15, 25, and 64 of the present algorithm with contour points (particles) shown in white. The correct number of 5 connected FG regions is found. (i) Energy evolution for both algorithms. For the present algorithm we show $\mathcal{E}_{\mathrm{length}}$ (dash-dotted), $\mathcal{E}^{\mathrm{PC}}_{\mathrm{data}}$ (dashed), and the total energy (sold). Circles mark region-fusion events. The red line shows the GC energy evolution for an initial $M = 12$. Crosses mark iterations. The residual energy of the ground-truth image is indicated by the horizontal dashed blue line.

Fig. 5. Visual comparison on natural-scene images using $\mathcal{E}^{\mathrm{PC}}$. (a/c) Segmentation result using the present algorithm; (b/d) using GC. GC finds 3 regions in (b) and 4 in (d). The present algorithm finds 3 connected FG regions in (a) and 9 in (c).

C. Multi-region piecewise smooth image model

1) External energy: Larger objects in images frequently have an inhomogeneous intensity distribution. This requires a model that allows for piecewise smooth representations of the image, such as the MS functional. Brox and Cremers have shown [35] that the MS functional is a first-order approximation to a Bayesian posterior maximizer where the likelihood considers region statistics over local windows. We therefore approximate the piecewise smooth MS model by overlapping piecewise constant patches within each region. We use spherical patches centered at each particle. Patches at interior pixels are not required.



(a) (b) (c) (d)

Fig. 6. Real-world application using $\mathcal{E}^{\text{PC}}$ to segment nuclei in a zebrafish embryo imaged by confocal fluorescence microscopy. (a) Visualization of the nuclei in the raw 3D data (image: Dr. Andrew Oates and Bhavna Rajasekaran, MPI-CBG Dresden). (b) Maximum-intensity projection of the final label image L. The algorithm is initialized with small FG regions placed at all local intensity maxima after Gaussian ($\sigma_x = 5\,\text{px}$, $\sigma_y = 5\,\text{px}$, $\sigma_z = 2\,\text{px}$) blurring. The topology is fixed to the initial topology, with the exception that regions are allowed to vanish. On average $1.03 \cdot 10^6$ candidate particles are processed per iteration. 99.99% of the particles stop moving after 25 iterations. The algorithms converges after 44 iterations finding 3218 connected FG regions. Since every connected component is a separate region with its own intensity estimate, nuclei of different brightnesses (e.g., arrows A and B) are correctly segmented. (c) Magnified z-plane showing an overlay of the original image with the final contours (black) in the region highlighted in yellow in (b) (intensities inverted for display purposes only). Touching nuclei are not fused if region merges are disallowed during contour evolution. (d) Allowing regions to merge, touching nuclei of similar intensities are assigned to the same region (e.g., arrow C) and the final number of connected FG regions is 1452. The visualizations in (a) and (b) were done using Imaris by Bitplane, Inc.

All statistics of the propagating regions are then computed locally per patch. This leads to the external energy

$$\mathcal{E}^{\text{PS}}_{\text{data}} = \sum_{i=0}^{M-1} \sum_{x \in X_i} \left(\sum_{y \in S^R_x \cap X_i} \frac{I(y)}{|X_i \cap S^R_x|} - I(x) \right)^2 , \tag{5}$$

with R the radius of the spherical patches and S^R_x the hypersphere of radius R centered at x. Within each patch S^R_x the intensity is constant. Smaller R hence lead to better representation of intensity gradients within regions, at the cost of reduced minimization robustness. The smaller R, the closer the initial segmentation needs to be to the final result.

In order to render the contour evolution more robust with respect to the initial segmentation, we propose adding the data-dependent balloon energy

$$\mathcal{E}_{\text{balloon}} = I \cdot H(-L + 1) \, . \tag{6}$$

This generates an outward flow whose strength depends on the image intensity. This flow counteracts the curvature-regularization flow in a data-dependent manner. The external energy for the pieceweise smooth case hence is $\mathcal{E}^{\text{PS}}_{\text{data}} + \beta \mathcal{E}_{\text{balloon}}$.

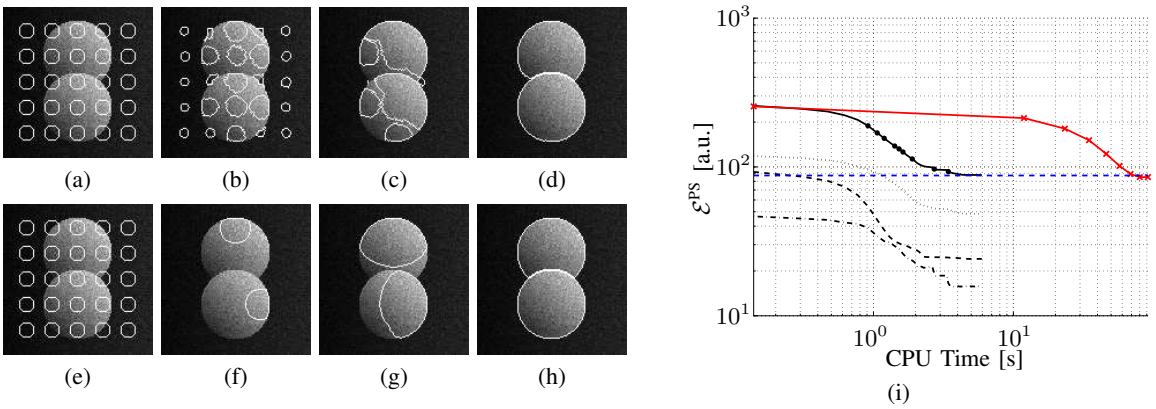


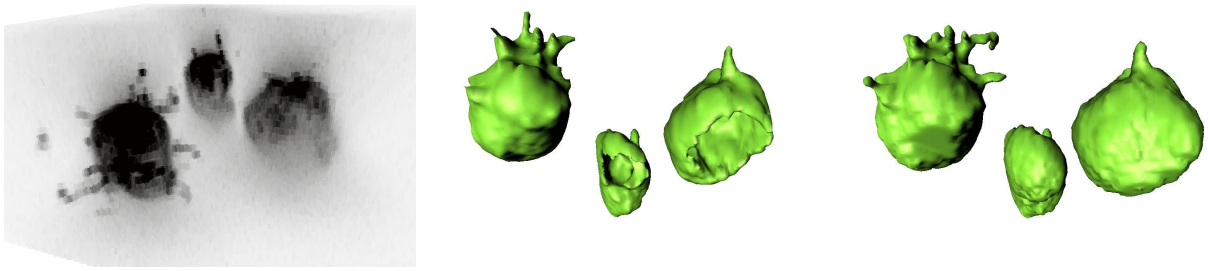
Fig. 7. Synthetic example using the energy $\mathcal{E}^{\mathrm{PS}}$. Two overlapping linearly shaded circles on a linearly shaded background, corrupted with Poisson noise. The brighter parts of the circles (top right) approximately correspond to a peak SNR of 8.7, while the low-intensity parts (bottom left) have SNR ≈ 3.2. (a–d) Contour evolution at iterations 0, 5, 15, and 70 of the present algorithm. The correct number of 2 connected FG regions is found. (e–h) Evolving contour at iterations 0, 1, 4, and 9 of the GC algorithm, also finding the correct number of regions. (i) Energy evolution for the two algorithms. For the present algorithms we show the evolution of $\mathcal{E}^{\mathrm{PS}}_{\mathrm{data}}$ (dashed), $\mathcal{E}_{\mathrm{length}}$ (dash-dotted), $\mathcal{E}_{\mathrm{balloon}}$ (dotted), and of the total energy $\mathcal{E}^{\mathrm{PS}}$ (solid). Circles mark region-fusion events. The red line shows the energy for GC. Crosses mark iterations. The residual energy of the ground-truth image is indicated by the horizontal dashed blue line.

We also adapt the region-merging criterion to only rely on local statistics: the empirical distributions P_{X_i} and P_{X_j} in Eq. 3 are only computed over the spherical mask S_x^R, as $P_{X_i \cap S_x^R}$ and $P_{X_j \cap S_x^R}$. This prevents merging regions that are separated by a large intensity gradient, even if they globally share similar empirical distributions (see Fig. 7d). For efficiency, P can be computed along with $\Delta\mathcal{E}^{\mathrm{PS}}_{\mathrm{data}}$.

2) Implementation: A neighborhood of size $O(R^d)$ needs to be read from the images I and L for every evaluation of the energy functional. Double lookups are avoided by computing the statistics in S_x^R along with the curvature flow. This results in an overall computational complexity in $O(|\Gamma|R^d)$ per iteration.

3) Benchmarks on synthetic data: Figure 7 illustrates the behavior of the present algorithm (Figs. 7a to 7d) on an image with linearly shaded FG and BG and compares it to GC (Figs. 7e to 7h). In the high-SNR areas, the data term of the energy dominates the evolution, and the contours immediately stick to intensity edges. Within the shaded FG circles, the regions expand as driven by the balloon force. After 5 iterations, regions that are not separated by large intensity gradients begin to merge.

The present algorithm is robust with respect to different choices of the patch radius R. However, R should be chosen smaller than the length scale of intensity variations and large enough such that $|S^R|$



(a) (b) (c)

Fig. 8. Real-world application using $\mathcal{E}^{\text{PS}}$ to segment primordial germ cells in a zebrafish embryo. (a) The raw 3D confocal image showing 3 cells with a fluorescent membrane stain (image: Mohammad Goudarzi, University of Münster). Intensities are inverted for display purposes only. (b) Intensity isocontour illustrating the inhomogeneity of the objects (bottom view). (c) Final segmentation using the present algorithm with $\mathcal{E}^{\text{PS}}$ (bottom view). The algorithm is initialized with a single box-shaped contour encompassing all objects and ultimately finds 3 connected FG regions. Visualizations were done using Imaris by Bitplane, Inc.

(a) (b) (c) (d)

Fig. 9. Visual comparison on natural-scene images using $\mathcal{E}^{\text{PS}}$. (a/c) Segmentation result using the present algorithm; (b/d) using GC. GC finds 6 regions in (b) and 9 in (d). The present algorithm finds 17 connected FG regions in (a) and 14 in (c).

constitutes a representative sample to construct the local intensity histograms P.

Figure 7i shows the evolution of all energy terms for the present example. When initialized with 25 bubbles as shown, GC is about 20 times slower than the present algorithm since it evaluates the energy everywhere in the image, whereas the present algorithm evaluates it only on the particles. Both methods find solutions close to ground truth and correctly estimate the number of regions.

The results for a 3D version of the image in Fig. 7 are given in Table I. In the 3D case, GC is initialized with the ground-truth number of regions and an initial contour close to the ground-truth solution in order to keep CPU times reasonable. The present algorithm is again initialized with bubbles.

4) Application to real data: Real-world applications of the present image model are shown in Figs. 8 and 9. The data consist of a 3D confocal image of primordial germ cells in a zebrafish embryo (Fig. 8a) and 2D natural-scene images from the Berkeley database (Fig. 9) [34]. The difficulty in segmenting these images is that the intensity is inhomogeneous within each object, as illustrated in Fig. 8b. Also

the background is heavily inhomogeneous in all images, requiring a piecewise smooth model. The final segmentations obtained with the present algorithm are shown in Figs. 8c, 9a, and 9c. The segmentations using GC are shown in Figs. 9b and 9d. Comparing Figs. 9c/d with Figs. 5c/d illustrates the difference between a piecewise constant and a piecewise smooth image model.

D. Multi-region deconvolving image model

1) External energy: The process of image acquisition maps the light irradiance of a real-world scene to a scalar field in Ω. This mapping is often modeled by its impulse response function, the *point-spread function* (PSF). Most notably in microscopes and telescopes, the mapping is largely linear, with nonlinear imaging effects playing a subordinate role. Image formation in these cases can hence be modeled as a (discrete) convolution of the real-world scene with the PSF. The result is corrupted by a pixel-wise noise process [8], [9]. Frequently, one is interested in reconstructing the shapes of the imaged real-world objects from the observed image, attempting to undo the PSF mapping. This is an inverse problem and the presence of noise renders its direct solution infeasible. The process of solving a regularized version of this inverse problem is often referred to as *deconvolution*, and multiple regularization methods are available [36]–[38]. In deconvolving active contours [9], the image model and the evolution of the contour serve as a natural regularization for the deconvolution. Moreover, the actual inverse problem never needs to be computed, since forward convolution is sufficient to evaluate the model energy. This has enabled highly accurate and robust reconstructions of small, diffraction limited objects in biological cells using fluorescence microscopy [39]. Here we extend the concept of deconvolving active contours to higher-dimensional images and to multiple regions, the number of which does not need to be known *a priori*.

Assuming that the noise process in the image-formation model follows a Gaussian distribution, the maximum-likelihood solution of the deconvolution problem is found by minimizing the energy functional

$$\mathcal{E}_{\text{data}}^{\text{dec}} = \sum_{x\in\Omega} \left(c_0 + \left(\sum_{i=1}^{M-1} c_i O_i(x) \right) * \text{PSF}(x) - I(x) \right)^2, \tag{7}$$

where c_i is the difference between the estimated intensity in FG region i and the BG intensity c_0, O_i the indicator function of region i, PSF the point-spread function of the imaging device, and I the observed image. This model assumes that the intensities c_i are constant within regions.

2) Implementation: Naive evaluation of the energy difference at a particle p requires two local convolutions around x_p. This can be avoided by introducing the *model image* $J = c_0 + \left(\sum_{i=1}^{M-1} c_i O_i \right) * \text{PSF}$. This model image is pre-computed using FFT at the beginning of each iteration. When a particle at position x

changes from region i to region j, the binary indicator O_i is updated to $O_i - \delta_x$ and O_j becomes $O_j + \delta_x$, where δ_x is the Kronecker delta (unit impulse) at x. Due to the linearity of the convolution operator:

$$\left(\sum_{k=1}^{M-1} O_k c_k + \delta_x c_j - \delta_x c_i\right) * \text{PSF} = \underbrace{\left(\sum_{k=1}^{M-1} O_k c_k\right) * \text{PSF}}_{\text{J}-c_0} + (\delta_x c_j - \delta_x c_i) * \text{PSF}\,. \tag{8}$$

The first term on the right-hand side corresponds to the pre-computed model image J without the BG. The second term is a scaled, discretized PSF mask. The model image J is then updated as $\widetilde{J}_x = J + \delta_x * \text{PSF} \cdot \frac{c_j}{c_i}$. Hence, Eq. 8 allows computing $\Delta\mathcal{E}^{\text{dec}}_{\text{data}}$ as a *local* operation: We iterate through a local window (centered at x_p) with a radius ρ equal to the PSF support. At each pixel y in the window we calculate $\widetilde{J}_{x_p}(y)$ and sum the quadratic differences to form $\Delta\mathcal{E}^{\text{dec}}_{\text{data}}(x_p) = \sum_y \left(I(y) - \widetilde{J}_{x_p}(y)\right)^2 - \sum_y (I(y) - J(y))^2$. After updating c_i, the entire model image J is re-computed from its definition. The computational complexity of the overall algorithm hence is in $O(|\Omega| \log |\Omega| + |\Gamma|\rho^d)$, where $|\Omega|$ is the total number of pixels in the image. Unlike for the previous energy functionals, the computational cost here depends on the size of the image, due to the convolution for computing J.

3) Intensity estimation: Estimating the region intensities c_i requires special attention in the present image model, particularly for objects that are small compared to the width of the PSF. We perform alternate minimization of the energy functional with respect to the contour shape and the estimated region intensities. The latter is done for fixed L (and hence O). We then estimate the intensities c_i such that the current model image J minimizes the L^2-distance to the data image I. We do this by formulating the problem as a 2D linear regression for each FG region: Considering the pixels within a region i as the data points, we find an affine transform of the set $A = \{I(x) : |L(x)| = i\}$ to the set $B = \{J(x) : |L(x)| = i\}$. We hence minimize $\sum_{i=0}^{M-1} (\vec{w} \cdot [1, A_i]^T - B_i)^2$ with respect to $\vec{w}$. The regression coefficient w_0 then serves as an estimate for the BG intensity, while w_1 is used as a correction factor for the current FG intensity estimates, hence $c_i \leftarrow w_1 c_i$.

4) Benchmark on synthetic data: Figure 10 illustrates the behavior of the present algorithm using the deconvolving energy functional on a synthetic image. The image simulates a realistic scenario in fluorescence microscopy with a pixel size of 80 nm and a half-width of the PSF of 120 nm. The image as blurred by the PSF (Fig. 10b) is corrupted with Poisson noise (Fig. 10c) with a peak SNR of 3 and 4 for the dimmer and brighter object, respectively. The width of the gap between the objects is equal to the half-width of the PSF.

Without using the information of how many objects are represented in the image, we start the segmentation from a single, rectangular initial contour (Fig. 10f). Figures 10f to10j show the evolution of the

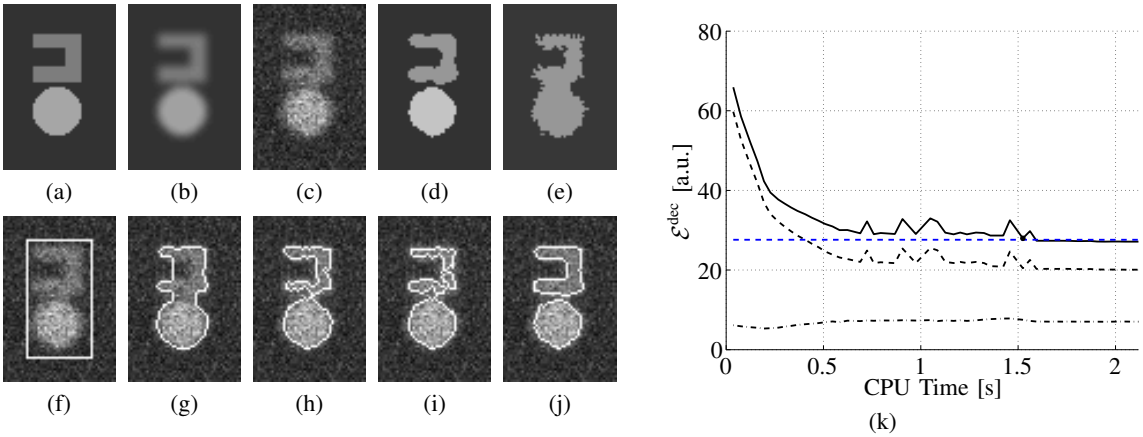
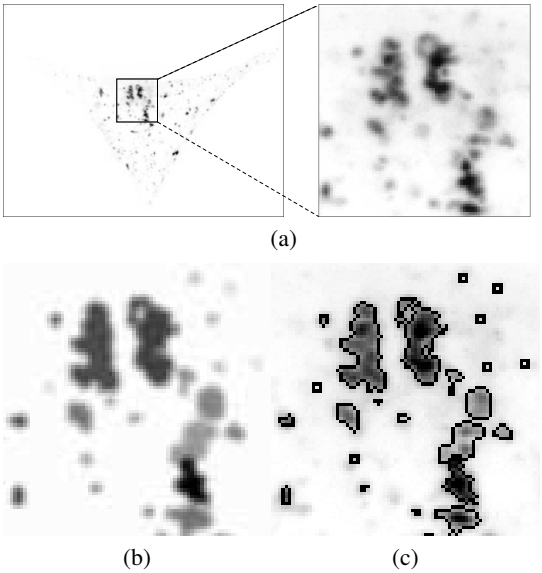


Fig. 10. Synthetic example using the energy $\mathcal{E}^{\text{dec}}$. (a) The ground-truth image. (b) The image convolved with a Gaussian point-spread function with $\sigma = 1.75\,\text{px}$, modeling a confocal fluorescence microscope. (c) The blurred image after addition of Poisson noise. The intensity of the u-shaped object corresponds to a peak SNR of 3, that of the circular region to an SNR of 4. (d) The reconstructed image using the deconvolving model. (e) The reconstructed image using a piecewise constant model with $\lambda = 0.1$, $\theta = 0.8$. (f–j) The contour after 1, 10, 20, 35, and 53 iterations, finding the correct number of 2 connected FG regions. (k) Energy evolution for the deconvolving model. The solid line represents the total energy, the dashed line $\mathcal{E}^{\text{dec}}_{\text{data}}$, and the dash-dotted line $\mathcal{E}_{\text{length}}$. The circle symbol indicates a region-merging event. The residual energy of the ground-truth image is indicated by the horizontal dashed blue line.

contour. Since the area of the circle is larger than the area of the u-shaped object, the intensity estimate is initially dominated by the circle. This causes initial over-segmentation of the u-shaped object. At iteration 19, the lower region splits into two regions with independent intensity estimates. This causes the regions segmenting the u-shaped object to merge again, resulting in a correct detection in the end. Figure 10k shows the evolution of the energies during this segmentation process.

We compare the results with those obtained using the piecewise constant energy without deconvolution (Eq. 4). The corresponding final reconstruction is shown in Fig. 10e. The PC model is not able to separate the two objects. It is moreover necessary to set λ to be 10 times larger than for the deconvolving energy in order to prevent overfitting the blurry object boundaries with many small regions.

5) Application to real data: The deconvolving energy functional is particularly useful when segmenting near-diffraction-limited objects as they occur, e.g., in intra-cellular imaging. We illustrate this in Fig. 11 using a single plane of a 3D confocal image showing endosomes labeled with fluorescent Rab5 protein [39]. Endosomes are small membrane-bound organelles of about 20 to 200 nm size. Accurately reconstructing the outlines of the many blurred, dense objects in this image is challenging when not accounting for the microscope PSF. Here we use a simple Gaussian model PSF whose width is determined by fitting



(a) (b) (c)

Fig. 11. Real-world application of the deconvolving model to fluorescently labeled endosomes in live HER911 cells. (a) Confocal fluorescence microscopy image after background subtraction using a rolling-ball algorithm (image: Prof. Urs Greber, University of Zurich, and Dr. Christoph Burckhardt, Harvard University). (b) Final reconstructed image in the inset window shown in (a). (c) The final contours (black pixels) overlaid onto the original image data. Starting from 1541 spherical FG regions centered at local intensity maxima, the algorithm finds 72 connected FG regions. We approximate the PSF by a Gaussian with $\sigma = 1.011$ px, found by fitting to signals of point-like structures in the image. Intensities are inverted for display purposes only.

it to point-like structures in the image. A separate measurement of the actual PSF of the microscope was not performed. Initially, we place small circular contours around every local intensity maximum in the image. These contours then rapidly evolve to concentrate around the endosomes. The number of regions in the image does not need to be known when initializing the algorithm. This is an advantage over explicit deconvolving active contours [9]. Explicit deconvolving active contours, however, provide sub-pixel resolution, whereas the present method is limited to pixel-level accuracy. This prevents the correct detection of objects covering less than 2 pixels. After 73 iterations, the algorithm converges to the reconstructed model image shown in Fig. 11b. The original image overlaid with the final outlines in the region indicated in Fig. 11a is shown in Fig. 11c. The two touching objects in the lower-right corner are properly separated based on their different intensities.

V. Discussion and Conclusions

We have presented a discrete multi-region-competition framework based on the topological constraint that each foreground region has to correspond to a connected set of pixels in some discrete geometry representation. An energy-minimization algorithm that accounts for this topological constraint has been implemented in both 2D and 3D and tested using three popular energy functionals. The number of regions in an image does not need to be known *a priori*, and the initial segmentation can have a different topology than the final result. We have presented a novel discrete contour propagation scheme and adapted concepts from digital topology to multiple regions in order to enforce the topological region definition and provide optional control over region merging and splitting during contour evolution. The contours are represented

by computational particles that evolve as driven by the energy-minimizing flow. Like discrete level-set methods [23], the present algorithm only requires evaluations of the energy functional, but not of its gradient. This is beneficial given the non-differentiable topological constraint. Contour oscillations are suppressed by adaptive step size reduction in the rank-based minimization algorithm.

We illustrated the algorithm on synthetic images and demonstrated its applicability to real-world data using three different energy functionals. We compared with results obtained using a state-of-the-art discrete energy minimizer based on multi-label graph cuts (GC) [11]. The first energy represented a piecewise constant image intensity model, extending the Chan-Vese model [3] to multiple foreground regions. The second functional used a piecewise smooth image model to allow for inhomogeneous intensity distributions within regions. This was done using local window statistics and an additional intensity-scaled balloon flow to improve the robustness with respect to the initial segmentation. The third energy functional included a convolution kernel to model the transfer function of an imaging device. This unites image deconvolution and segmentation and extends explicit deconvolving active contours [9] to handle topological changes during energy minimization and to higher-dimensional images. The benchmarks demonstrated that the solution quality and the runtime of the present algorithm are competitive. Compared with GC, the present algorithm is particularly beneficial for large numbers of regions and for costly energy functionals, such as the approximated piecewise smooth energy.

Due to the discrete contour representation, the present method is limited to single-pixel accuracy. Sub-pixel accurate segmentations, such as those achieved by explicit deconvolving active contours [9] would require continuously varying particle positions, hampering the efficient solution of the energy minimization problem and the application of digital topology. A limitation of the present method compared to GC is that contours can advance at most one pixel per iteration. For initial contours far from the final solution, segmentation may hence be slow. Nevertheless, the timings of the present implementation as reported for each test case are encouraging when compared with GC. The computational cost of the algorithm depends on the energy functional to be minimized. In the example of Fig. 8, evaluations of the energy functional accounted for 88% of the computational time (66% for $\mathcal{E}_{\text{data}}^{\text{PS}}$, 22% for the curvature-regularizing flow), whereas topology processing took 1%, contour propagation 4%, and data-structure update 3%. Table II shows this breakdown of the computational cost for each of the three energy functionals considered. For the PC model, curvature approximation and contour propagation are the most expensive parts. This is due to lookups in L and I. For the same reason, the computational time using the PS model is dominated by evaluating the data energy. For the deconvolving energy functional, pre-computing the model image J dominates the processing time. The time complexity of the algorithm

TABLE II

RELATIVE COMPUTATIONAL COSTS OF THE DIFFERENT STEPS OF THE ALGORITHM FOR THE THREE ENERGY FUNCTIONALS (PC, PS, DEC) CONSIDERED HERE. ALL TIMES WERE MEASURED USING THE RESPECTIVE BIOLOGICAL EXAMPLE IMAGES.

	evaluating $\mathcal{E}_{\text{data}}$	evaluating $\mathcal{E}_{\text{length}}$	optimization	contour propagation	topology processing	data-structure update
PC	1%	**31**%	21%	**31**%	4.5%	11.5%
PS	**66**%	22%	4%	4%	1%	3%
DEC	**97**%	<1%	2%	<1%	<1%	<1%

with the PC and PS image models is linear in the total number of particles, i.e., the total contour length, and is independent of the image size. For the deconvolving energy functional, however, the convolution renders the complexity dependent on the image size.

The computational performance of the present method could be further improved in a number of ways. Storing the image data along a space-filling curve is expected to improve cache efficiency, as points that are close in the image will also be close in memory. Future work will also explore the possibility of computing the energy differences of different particles in parallel, using multi-threading or graphics processing units (GPU). In addition, we are currently extending the present framework to include particle–particle interaction potentials as additional regularization [40].

The presented algorithm has been implemented as an image filter in the Insight Toolkit (ITK) image-processing library [25] and is available as open source from the authors.

ACKNOWLEDGMENTS

We thank Dr. Jo A. Helmuth for sharing his experience in many discussions and Prof. U. Greber (University of Zurich), Dr. Ch. Burckhardt (Harvard University), Dr. P. Liberali (University of Zurich), Dr. A. Oates and B. Rajasekaran (both MPI-CBG Dresden), and M. Goudarzi (University of Münster) for kindly having provided example images. JC and GP were funded through CTI grant 9325.2-PFLS-LS from the Swiss Federal Commission for Technology and Innovation (to IFS in collaboration with Bitplane, Inc). This project was further supported with grants from the Swiss SystemsX.ch initiative, grants LipidX and WingX, to IFS.

REFERENCES

[1] X. Han, C. Xu, and J. L. Prince, "A topology preserving level set method for geometric deformable models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, pp. 755–768, 2003.

[2] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Communications on Pure and Applied Mathematics*, vol. 42, no. 5, pp. 577–685, 1989.

[3] T. F. Chan and L. A. Vese, "Active contours without edges," *Image Processing, IEEE Transactions on*, vol. 10, no. 2, pp. 266 –277, feb 2001.

[4] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, pp. 1222–1239, 2001.

[5] Y. Boykov and V. Kolmogorov, "Computing geodesics and minimal surfaces via graph cuts," in *Computer Vision, IEEE International Conference on*, vol. 1. Los Alamitos, CA, USA: IEEE Computer Society, 2003, pp. 26–33.

[6] M. Nikolova, S. Esedoglu, and T. F. Chan, "Algorithms for finding global minimizers of image segmentation and denoising models," *SIAM Journal on Applied Mathematics*, vol. 66, no. 5, pp. 1632–1648, 2006.

[7] E. S. Brown, T. F. Chan, and X. Bresson, "Completely convex formulation of the Chan-Vese image segmentation model," *Int. J. Comput. Vis.*, 2011.

[8] F. Santosa, "A level-set approach for inverse problems involving obstacles," *ESAIM: Control, Optimisation and Calculus of Variations*, vol. 1, pp. 17–33, 1996.

[9] J. A. Helmuth and I. F. Sbalzarini, "Deconvolving active contours for fluorescence microscopy images," in *Proc. Intl. Symp. Visual Computing (ISVC)*, ser. Lecture Notes in Computer Science, vol. 5875. Las Vegas, USA: Springer, November 2009, pp. 544–553.

[10] T. Brox and J. Weickert, "Level set based image segmentation with multiple regions," in *Pattern Recognition*, ser. Lecture Notes in Computer Science, C. E. Rasmussen, H. H. Blthoff, B. Schlkopf, and M. A. Giese, Eds. Springer Berlin / Heidelberg, 2004, vol. 3175, pp. 415–423, 10.1007/978-3-540-28649-3_51.

[11] A. Delong, A. Osokin, H. N. Isack, and Y. Boykov, "Fast approximate energy minimization with label costs," *Int. J. Comput. Vis.*, 2011.

[12] B. Sandberg, S. H. Kang, and T. Chan, "Unsupervised multiphase segmentation: A phase balancing model," *Image Processing, IEEE Transactions on*, vol. 19, no. 1, pp. 119 –130, jan. 2010.

[13] T. Kadir and M. Brady, "Unsupervised non-parametric region segmentation using level sets," *Computer Vision, IEEE International Conference on*, vol. 2, p. 1267, 2003.

[14] L. A. Vese and T. F. Chan, "A multiphase level set framework for image segmentation using the Mumford and Shah model," *International Journal of Computer Vision*, vol. 50, pp. 271–293, 2002, 10.1023/A:1020874308076.

[15] A.-R. Mansouri, A. Mitchie, and C. Vazquez, "Multiregion competition: A level set extension of region competition to multiple region image partitioning," *Computer Vision and Image Understanding*, vol. 101, no. 3, pp. 137 – 150, 2006.

[16] S.-C. Zhu and A. Yuille, "Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation," *IEEE Transactions On Pattern Analysis and Machine Intelligence*, vol. 18, no. 9, pp. 884–900, Sep. 1996.

[17] J. Lie, M. Lysaker, and X.-C. Tai, "A variant of the level set method and applications to image segmentation," *Mathematics of Computation*, vol. 75, pp. 1155–1174, 2006.

[18] X. Fan, P.-L. Bazin, and J. Prince, "A multi-compartment segmentation framework with homeomorphic level sets," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, june 2008, pp. 1 –6.

[19] B. Song and T. Chan, "A fast algorithm for level set based optimization," *CAM-UCLA*, vol. 02-68, 2002.

[20] L. He and S. Osher, "Solving the Chan-Vese model by a multiphase level set algorithm based on the topological derivative," in *Proceedings of the 1st international conference on Scale space and variational methods in computer vision*, ser. SSVM'07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 777–788.

[21] I. Larrabide, R. Feijo, A. Novotny, and E. Taroco, "Topological derivative: A tool for image processing," *Computers & Structures*, vol. 86, no. 13-14, pp. 1386 – 1403, 2008, structural Optimization.

[22] L. Yu, Q. Wang, L. Wu, and J. Xie, "Mumford-Shah model with fast algorithm on lattice," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, vol. 2, May 2006, p. II.

[23] Y. Shi and W. Karl, "A real-time algorithm for the approximation of level-set-based curve evolution," *Image Processing, IEEE Transactions on*, vol. 17, no. 5, pp. 645 –656, May 2008.

[24] J. Malcolm, Y. Rathi, A. Yezzi, and A. Tannenbaum, "Fast approximate surface evolution in arbitrary dimension," in *Proc. SPIE 6914*, 2008.

[25] L. Ibanez, W. Schroeder, L. Ng, and J. Cates, *The ITK Software Guide*, 2nd ed., Kitware, Inc. ISBN 1-930934-15-7, http://www.itk.org/ItkSoftwareGuide.pdf, 2005.

[26] F. Segonne, "Segmentation of medical images under topological constraints," Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, MA, USA, 2006, aAI0809130.

[27] G. Bertrand, "Simple points, topological numbers and geodesic neighborhoods in cubic grids," *Pattern Recognition Letters*, vol. 15, no. 10, pp. 1003 – 1011, 1994.

[28] J. Lamy, "Integrating digital topology in image-processing libraries," *Computer Methods and Programs in Biomedicine*, vol. 85, no. 1, pp. 51 – 58, 2007.

[29] G. Bertrand, J. Everat, and M. Couprie, "Image segmentation through operators based on topology," *Journal of Electronic Imaging*, vol. 6, pp. 395–405, Oct. 1997.

[30] F. Calderero and F. Marques, "General region merging approaches based on information theory statistical measures," in *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, Oct. 2008, pp. 3016 –3019.

[31] I. Ayed and A. Mitiche, "A region merging prior for variational level set image segmentation," *Image Processing, IEEE Transactions on*, vol. 17, no. 12, pp. 2301 –2311, Dec. 2008.

[32] H. N. Isack and Y. Boykov, "Energy-based geometric multi-model fitting," *Int. J. Comput. Vis.*, 2011.

[33] J. Kybic and J. Kratky, "Discrete curvature calculation for fast level set segmentation," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*, Nov. 2009, pp. 3017 –3020.

[34] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *in Proc. 8th Intl Conf. Computer Vision (ICCV)*, Vancouver, BC, Canada, 2001, pp. 416–423.

[35] T. Brox and D. Cremers, "On the statistical interpretation of the piecewise smooth Mumford-Shah functional," in *Scale Space and Variational Methods in Computer Vision*, ser. Lecture Notes in Computer Science, F. Sgallari, A. Murli, and N. Paragios, Eds. Springer Berlin / Heidelberg, 2007, vol. 4485, pp. 203–213, 10.1007/978-3-540-72823-8_18.

[36] C. R. Vogel, *Computational Methods for Inverse Problems*. SIAM, 2002.

[37] J.-B. Sibarita, "Deconvolution microscopy," *Adv. Biochem. Eng. Biot.*, vol. 95, pp. 201–243, 2005.

[38] P. C. Hansen, J. G. Nagy, and D. P. O'Leary, *Deblurring Images: Matrices, Spectra, and Filtering*. SIAM, 2006.

[39] J. A. Helmuth, C. J. Burckhardt, U. F. Greber, and I. F. Sbalzarini, "Shape reconstruction of subcellular structures from live cell fluorescence microscopy images," *J. Struct. Biol.*, vol. 167, pp. 1–10, 2009.

[40] G. Sundaramoorthi and A. Yezzi, "Global regularizing flows with topology preservation for active contours and polygons," *Image Processing, IEEE Transactions on*, vol. 16, no. 3, pp. 803 –812, march 2007.

Janick Cardinale is a PhD student with the MOSAIC Group at ETH Zurich, Switzerland. He obtained his M.Sc. in Computer Science from ETH Zurich in 2007, majoring in Computational Science. His research interests are in image segmentation and quantification, recursive Bayesian filtering, and Markov chain Monte Carlo algorithms.

Grégory Paul is a postdoctoral researcher with the MOSAIC Group at ETH Zurich, Switzerland. He received his Ph.D. from the University of Paris VI in 2008. He holds a M.Sc. degree in Cell Biology and Physiology from École Normale Supérieure in Paris, which he completed in 2003. His research interests are in image processing and analysis, computational statistics, stochastic processes and algorithms, and theoretical biology.

Ivo F. Sbalzarini is Assistant Professor of Computational Science at ETH Zurich, Switzerland, and founder and head of the MOSAIC Group. He received his Ph.D. in Computer Science and his Diploma (equiv. M.Sc.) in Mechanical Engineering, majoring in Control Theory and Applied Mathematics, from ETH Zurich in 2006 and 2002, respectively. His research interests include image analysis and image processing, adaptive discretization schemes for PDEs, randomized algorithms, and parallel algorithms.